

REMARKS

Applicant is in receipt of the Office Action mailed January 22, 2010. Claims 2, 14, and 16 have been cancelled. Claims 1-13 and 15, and 17-28 have been amended. Claims 1-13 and 15-28 are pending in the case. Reconsideration of the present case is earnestly requested in light of the following remarks.

Section 101 Rejections

Claims 1-23 were rejected under 35 U.S.C. 101 as being directed to non-statutory subject matter, specifically, for reciting a “carrier medium”. Applicant has amended the claims accordingly, and respectfully requests removal of the section 101 rejection of the claims.

Section 102 Rejections

Claims 1-28 were rejected under 35 U.S.C. 102(b) as being anticipated by Kudukoli et al (US 20010020291, “Kudukoli”). Applicant has cancelled claims 2, 14, and 16, rendering their rejections moot.

Applicant respectfully reminds the Examiner that the standard for “anticipation” is one of strict identity. Anticipation requires the presence in a single prior art reference disclosure of each and every element of the claimed invention, arranged as in the claim. M.P.E.P 2131; *Lindemann Maschinenfabrik GmbH v. American Hoist & Derrick Co.*, 221 USPQ 481, 485 (Fed. Cir. 1984). The **identical** invention must be shown in as complete detail as is contained in the claims. *Richardson v. Suzuki Motor Co.*, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989).

Amended claim 1 recites:

1. A computer accessible memory medium comprising program instructions executable by a processor to:

receive a request for information regarding an interface of a graphical program;

programmatically determine the information regarding the interface of the graphical program, wherein the information regarding the interface of the graphical program includes information for invoking execution of the graphical program; and

return the information regarding the interface of the graphical program.

Applicant first notes that because claim 1 has nothing to do with programmatically generating a graphical program, nor a *user interface* of a generated graphical program, and because Kudukoli nowhere mentions or even hints at the claimed functionality regarding information for invoking execution of a graphical program, Kudukoli is not germane to claim 1. In other words, Kudukoli is non-analogous art with respect to claim 1. More detailed analysis is presented below.

Nowhere does the cited art teach **receive a request for information regarding an interface of a graphical program**, as recited in claim 1. As recited in the claim, **the information regarding the interface of the graphical program includes information for invoking execution of the graphical program**.

Kudukoli is directed to programmatic generation of a graphical program (via a graphical program generation program (GPG), and cited paragraph 0036 in particular discusses an application programming interface (API) for programmatically adding graphical program components to a graphical program. Applicant respectfully notes that the cited interface (API) is not the interface for the graphical program it is used to create, and further, that the cited “program information” referred to in the Abstract of Kudukoli specifies *functionality of a graphical program (or portion) to be created*, and thus is specifically *not* information regarding an interface of the graphical program. Nor does this citation, nor Kudukoli in general, describe “receiving a request” for such information.

Cited paragraph 0083, originally directed to claim 2, now cancelled, and whose subject matter has been added to claim 1, discloses various exemplary applications for the GPG program and the generated graphical program, e.g., “data acquisition/generation, analysis, and/or display, and for controlling or modeling instrumentation or industrial automation hardware”, and mentions client/server program implementations, but nowhere

discusses or even mentions information regarding an interface for invoking a graphical program, as recited in claim 1.

Thus, the cited art fails to disclose these features of claim 1.

Nor does the cited art teach **programmatically determine the information regarding the interface of the graphical program, wherein the information regarding the interface of the graphical program includes information for invoking execution of the graphical program**, as recited in claim 1.

Cited paragraph 0028 is directed to programmatic generation of a graphical program in response to user input specifying functionality, i.e., requirements, of the program, e.g., “user input specifying various display options desired”, or “user input specifying various analysis options desired”.

The Office Action argues that since the citation states that “the functionality of the graphical program is determined mainly by the GPG program that generates the graphical program, and only determined to a small degree (or no degree) by the received information”, this somehow teaches programmatic determination of information regarding an interface of the graphical program. Applicant submits that the determination of program *functionality* by the GPG program to programmatically *generate* the program is not equivalent to programmatically determining information regarding invocation/execution of the program.

Applicant respectfully notes that a) the cited information regarding display options and subsequent programmatic generation of a graphical program that displays a user interface, e.g., “to visually graph the numeric data”, is provided by the user, and is thus, not programmatically determined; and b) the claimed “interface” specifically refers to “information for invoking execution of the graphical program”, as recited in the amended claim, *not* to a user interface of the program for displaying data, as per the citation.

Applicant further respectfully notes that neither Applicant’s claims, nor the present Specification, describes the claimed “interface” as a user interface for displaying data or for user-interaction with the program, *see, e.g.*, in the first paragraph of the Summary (p.5, lines 3-9), which states “The interface information for the graphical

program may include information necessary to invoke execution of the graphical program. Thus, the interface information is also referred to herein as calling information. The interface information may include such information as parameters of the graphical program, their respective data types, whether each parameter is an input parameter, an output parameter, or both, and default values for input parameters.”, as well as p.6, lines 3-20, and p.23, line 12 – p.27, line 6, among others. While Applicant understands and appreciates the Examiner’s broad interpretation of this term (as referring to a *user interface* of the program), it is respectfully noted that this interpretation is contrary to the Specification and (dependent) claims, and thus improper. Applicant has amended the claims as indicated above to emphasize the proper meaning of the term.

Moreover, even under the Office Action’s (improper and incorrect) interpretation of this term, Applicant submits that the citation fails to teach “programmatically determine the information regarding the interface for the graphical program”, at least because per the cited text, Kudukoli’s information regarding display options is provided by the user, and is thus not determined programmatically.

Thus, the citation fails to disclose this feature of claim 1.

Nor does the cited art teach **return the information regarding the interface of the graphical program**, as recited in claim 1.

Cited paragraph 0124 discusses the relationship between *received* (from the user) program information used to generate a graphical program and the generated program, specifically, maintaining an association between the received information and the program so that subsequent iterations of the program may be generated, e.g., to resolve problems discovered in the initial implementation. Similar to the above, Applicant notes that the cited *received* program information is not programmatically determined, but rather, is provided by the user, and, more importantly, the cited *received* program information is used to *generate* Kudukoli’s graphical program, and thus is specifically *not* information for invoking execution of the program, as per claim 1.

Thus, the citation fails to disclose this feature of claim 1.

Thus, for at least the above reasons, Applicant submits that the cited art fails to teach or suggest all the features and limitations of claim 1, and so claim 1, and those claims respectively dependent therefrom, are patentably distinct and non-obvious over the cited art, and are thus allowable.

Independent claims 24 and 27 each includes similar limitations as claim 1, and so the above arguments apply with equal force to these claims. Moreover, cited paragraphs 0080, 0081, and 0082, also fail to disclose these features. Thus, for at least the above reasons, Applicant submits that claims 24 and 27, and those claims respectively dependent therefrom, are patentably distinct and non-obvious over the cited art, and are thus allowable.

Independent claim 15 also includes features and limitations regarding interface information, “wherein the information regarding the interface of the graphical program includes information for invoking execution of the graphical program”, which are neither taught nor suggested by Kudukoli.

For example, does the cited art teach **programmatically request information regarding an interface of a graphical program, wherein the information regarding the interface of the graphical program includes information for invoking execution of the graphical program, nor receive information regarding the interface of the graphical program in response to the request, nor invoke execution of the graphical program according to the received information regarding the interface of the graphical program**, as recited in claim 15.

Regarding the feature “programmatically request information regarding an interface of a graphical program, wherein the information regarding the interface of the graphical program includes information for invoking execution of the graphical program”, cited paragraph 0022 is directed to a GPG program that programmatically generates a graphical program based on functionality requirements provided by a user, but is silent regarding programmatically requesting information regarding an interface for a graphical program whereby the program can be invoked, i.e., a calling invocation interface, as claimed.

Cited paragraph 0027 discusses explicit specification of graphical program functionality by the user (based upon which, the GPG program programmatically

generates the program), but again, says nothing whatsoever regarding programmatic request for invocation or calling information of a graphical program.

Cited paragraph 0028, as discussed above, is directed to programmatic generation of a graphical program in response to *user input specifying functionality*, i.e., requirements, of the program, e.g., “user input specifying various display options desired”, or “user input specifying various analysis options desired”, where the functionality is specified implicitly, rather than explicitly. As explained above, the determination of program functionality by the GPG program to programmatically generate the program is not equivalent to programmatically determining information regarding invocation/execution of the program. Nor is receiving information implicitly specifying program functionality (for the purpose of generating the program) equivalent to requesting information regarding invocation of the program.

Thus, for at least these reasons, the citation fails to disclose this feature of claim 15.

Regarding the feature “receive information regarding the interface of the graphical program in response to the request”, cited paragraph 0098 is directed to a GPG program programmatically generating a graphical program (or portion) based on received information (from a user) regarding functionality requirements for the program, but is silent regarding receiving information regarding an interface for a graphical program whereby the program can be invoked, i.e., a calling invocation interface, as claimed.

Cited paragraph 0099 discusses the GPG processing the received information (specifying program functionality, either explicitly or implicitly) in order to determine how to generate the graphical program, but again, is silent regarding receiving information regarding a calling interface for a graphical program, as claimed.

Cited 0100 further discusses programmatic generation of the graphical program by the GPG program, including the GPG specifying “inclusion of various objects in the new graphical program”, such as nodes, user interface panels, interconnections between block diagram objects and/or user interface objects, and so forth, but says nothing about receiving information regarding the interface of the graphical program (for invoking the program), nor, more particularly, in response to requesting such information.

Thus, for at least these reasons, the citation fails to disclose this feature of claim 15.

Finally, regarding the feature “invoke execution of the graphical program according to the received information regarding the interface of the graphical program”, the cited Abstract and paragraph 0018 discuss programmatic generation of a graphical program by a GPG based on program information regarding desired functionality of the program received from a user, but makes no mention at all regarding invocation of a graphical program based on received information regarding a calling or invocation interface of the program.

Cited paragraph 0021 describes various different purposes or applications of the GPG program and its use, but again, makes no mention regarding invocation of a graphical program based on received information regarding a calling or invocation interface of the program, per claim 1.

Thus, for at least these reasons, the citation fails to disclose this feature of claim 15.

Thus, for at least these reasons, the citation fails to disclose all the features and limitations of claim 15, and so claim 15, and those claims respectively dependent therefrom, are patentably distinct and non-obvious over the cited art, and are thus allowable.

Independent claims 25 and 28 each includes similar limitations as claim 15, and so the above arguments apply with equal force to these claims. Moreover, cited paragraphs 0036, 0034, and 0237, as well as cited paragraphs 0080, 0081, and 0082, also fail to disclose these features. Thus, for at least the above reasons, Applicant submits that claims 25 and 28, and those claims respectively dependent therefrom, are patentably distinct and non-obvious over the cited art, and are thus allowable.

Independent claim 26 is also similar to independent claim 15, but where each action is performed by a test executive application. Applicant has read the citations presented by the Office Action closely, and respectfully submits that these citations, and Kudukoli in general, fail to teach or suggest these limitations, as discussed at length above. Thus, for at least these reasons, the citation fails to disclose all the features and limitations of claim 26, and so claim 26, and those claims respectively dependent

therefrom, are patentably distinct and non-obvious over the cited art, and are thus allowable.

Applicant asserts that numerous ones of the dependent claims recite further distinctions over the cited art.

For example, nowhere does the cited art teach **wherein said generating the data describing the data types of the parameters for invoking the graphical program comprises generating XML data describing the data types of the parameters for invoking the graphical program**, as recited in claim 6.

Cited paragraph 0026 mentions that program information provided by the user to the GPG program in the form of a state diagram (that specifies program functionality) may be received in any of various formats, including XML data, but says nothing about generating XML data describing the data types of the parameters for invoking the graphical program, nor generating data describing the data types of parameters for invoking a graphical program at all, and thus does not, and cannot, teach these features.

Applicant also asserts that numerous other ones of the dependent claims recite further distinctions over the cited art. However, since the independent claims have been shown to be patentably distinct, a further discussion of the dependent claims is not necessary at this time.

Removal of the section 102 rejection of the claims is earnestly requested.

CONCLUSION

Applicant submits the application is in condition for allowance, and an early notice to that effect is requested.

If any extensions of time (under 37 C.F.R. § 1.136) are necessary to prevent the above-referenced application(s) from becoming abandoned, Applicant(s) hereby petition for such extensions. The Commissioner is hereby authorized to charge any fees which may be required or credit any overpayment to Meyertons, Hood, Kivlin, Kowert & Goetzel P.C., Deposit Account No. 50-1505/5150-75401/JCH.

Also filed herewith are the following items:

- ☐ Request for Continued Examination
- ☐ Terminal Disclaimer
- ☐ Power of Attorney By Assignee and Revocation of Previous Powers
- ☐ Notice of Change of Address
- ☐ Other:

Respectfully submitted,

/Jeffrey C. Hood/

Jeffrey C. Hood, Reg. #35198
ATTORNEY FOR APPLICANT(S)

Meyertons, Hood, Kivlin, Kowert & Goetzel PC
P.O. Box 398
Austin, TX 78767-0398
Phone: (512) 853-8800
Date: 2010-02-23 JCH/MSW